

```

;+
;***** USER-DEFINED function "READ_CSV_FILE" *****
;
;Author:      wk
;Creation Date: 02-15-02
;
;PURPOSE:
; this IDL routine will read any comma delimited text file and return a 2D text array that user can easily
; change format later on
; Main purpose is to read text file generated by GIS
;
;NOTE:
; have to have one header line and only one!!
;
;SYNTAX:
; result = read_csv_file(<filename>)
;-
;2010.09.02 Modified to allow for longer strings to be read in (previously only 99 characters allowed)
function read_csv_file,filename, NOHEADER=noheader

;Read text file into a string array that size = # rows
openr,U,filename,/get_lun
nobs=file_lines(filename)    ;# of records
if not keyword_set(noheader) then begin
  readf,U,format='(/)' ;Skip header
  nobs=nobs-1
endif
dummy=strarr(nobs)
readf,U,dummy  ;read data
free_lun,U

;Determine # columns in one row (using the 1st row)
icount=0
istart=0
ipos=0
chkstring=trim(dummy[0])

while ipos ne -1 do begin
  ipos=strcmp(chkstring,",",istart)
  istart=ipos+1
  icount=icount+1
endwhile

;initial arrays
colm =icount  ;# columns in one raw
arr =strarr(colm,nobs)

for j=0L,nobs-1L do begin

  chkstring=trim(dummy[j])

  istart=-1  ;assum a comma before position 0
  for i=0,colm-1 do begin

```

```
iend=strpos(chkstring,",",istart+1)
length=(iend-1)-istart
if iend eq -1 then length=200 ; assume the last column length=99

arr[i,j]=strmid(chkstring,istart+1,length)
istart=iend ;restart at the current comma position
endfor
endfor

return,arr

end
```